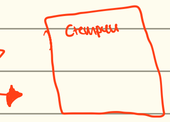
Introduction to HTML.

24/1/25

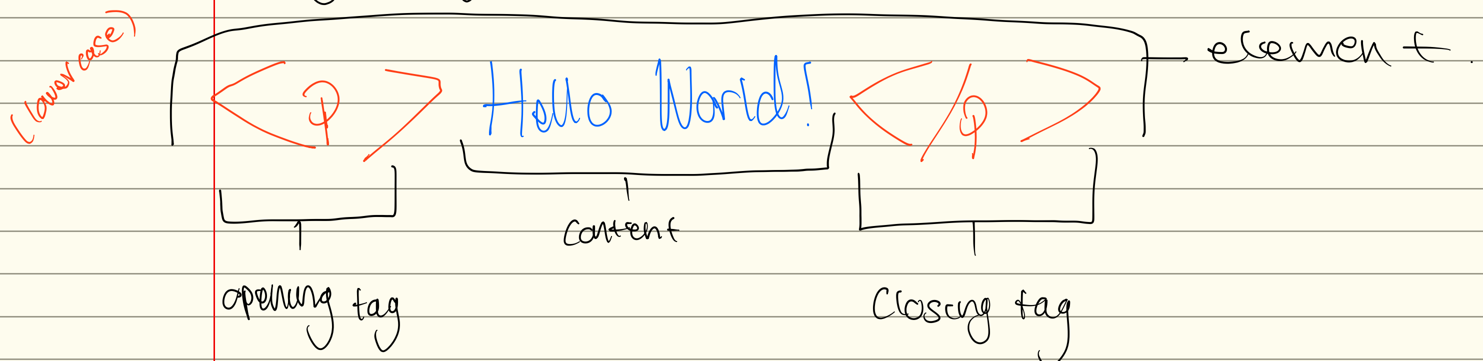
- HTML provides structure to the content appearing on a website, such as images, text or videos.
- HTML stands for HyperText Markup Language:
  - A markup language is a computer language that defines the presentation of raw text.
  - In HTML, the computer can interpret raw text that is wrapped in HTML elements.
  - HyperText is text displayed on a computer device that provides access to other text through links, also known as hyperlinks.

<h1> ctempuu</h1>



HTML Anatomy

- HTML is composed of elements. These elements structure the webpage & define its content.



The Body

<Body> </Body> → Main body text.

<p> </p> → Paragraph.

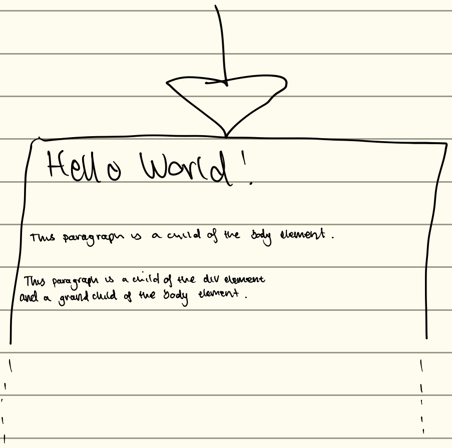
HTML Structure.

- HTML is organised as a collection of family tree relationships. When an element is contained inside another element it's considered the child of that element.
  - ↳ The child element is nested inside of the parent element.

```
<body>
    <p> This paragraph is the child
of the body </p>
</body>
```

- There is multiple levels of nesting, this analogy can be extended to grandchildren, great grandchildren, etc.
- The relationship between elements & their ancestor & descendent elements is known as hierarchy.

```
<body>
    <h1> Hello World! </h1>
    <p> This paragraph is the child of
    the body element. </p>
    <div>
        <p> This paragraph is a child of the div
        element and a grand child of the
        body element. </p>
    </div>
</body>
```


Hello World!
This paragraph is a child of the body element.
This paragraph is a child of the div element and a grand child of the body element.

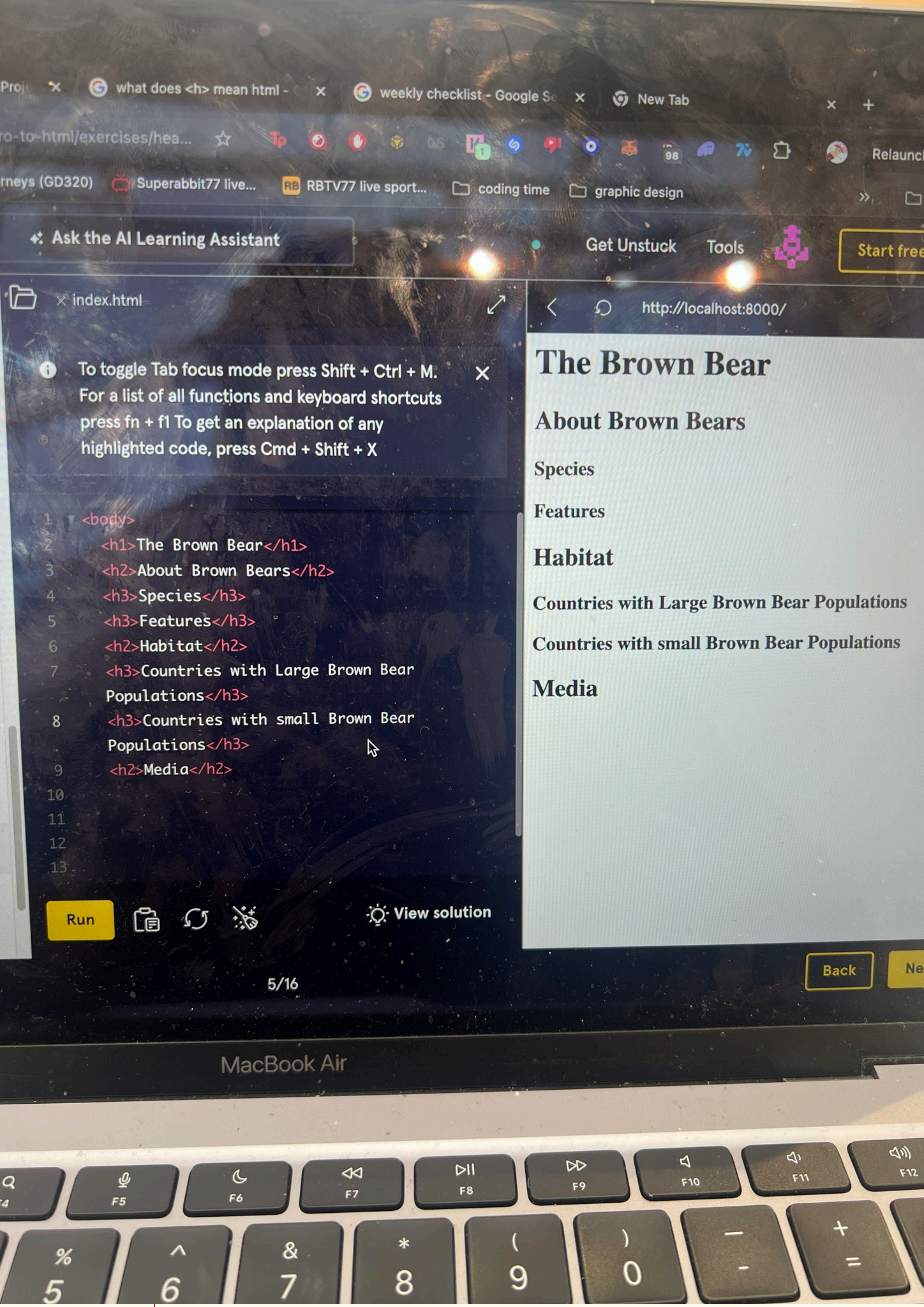
Headings `<h1> </h1>`

- There are 6 different headings / heading elements. Headings can be used for a variety of purposes like titling sections, articles, or other forms of content.
- The list of heading elements available in HTML.

- `<h1>` - used for main headings. All other smaller headings are used for subheadings.
- `<h2>`
- `<h3>`
- `<h4>`
- `<h5>`
- `<h6>`

```
<h1> BREAKING NEWS </h1>
```

↳ used largest heading available to catch readers attention.


Proj
what does <h> mean html -
weekly checklist - Google Se
New Tab
ro-to-html/exercises/hea...
Relaunc
rneys (GD320)
Superabbit77 live...
RBTV77 live sport...
coding time
graphic design
Ask the AI Learning Assistant
Get Unstuck
Tools
Start free
index.html
http://localhost:8000/
Run
View solution
5/16
Back
Ne
MacBook Air

To toggle Tab focus mode press Shift + Ctrl + M. For a list of all functions and keyboard shortcuts press fn + f1 To get an explanation of any highlighted code, press Cmd + Shift + X

```
<body>
  <h1>The Brown Bear</h1>
  <h2>About Brown Bears</h2>
  <h3>Species</h3>
  <h3>Features</h3>
  <h2>Habitat</h2>
  <h3>Countries with Large Brown Bear Populations</h3>
  <h3>Countries with small Brown Bear Populations</h3>
  <h2>Media</h2>
```

The Brown Bear

About Brown Bears

Species

Features

Habitat

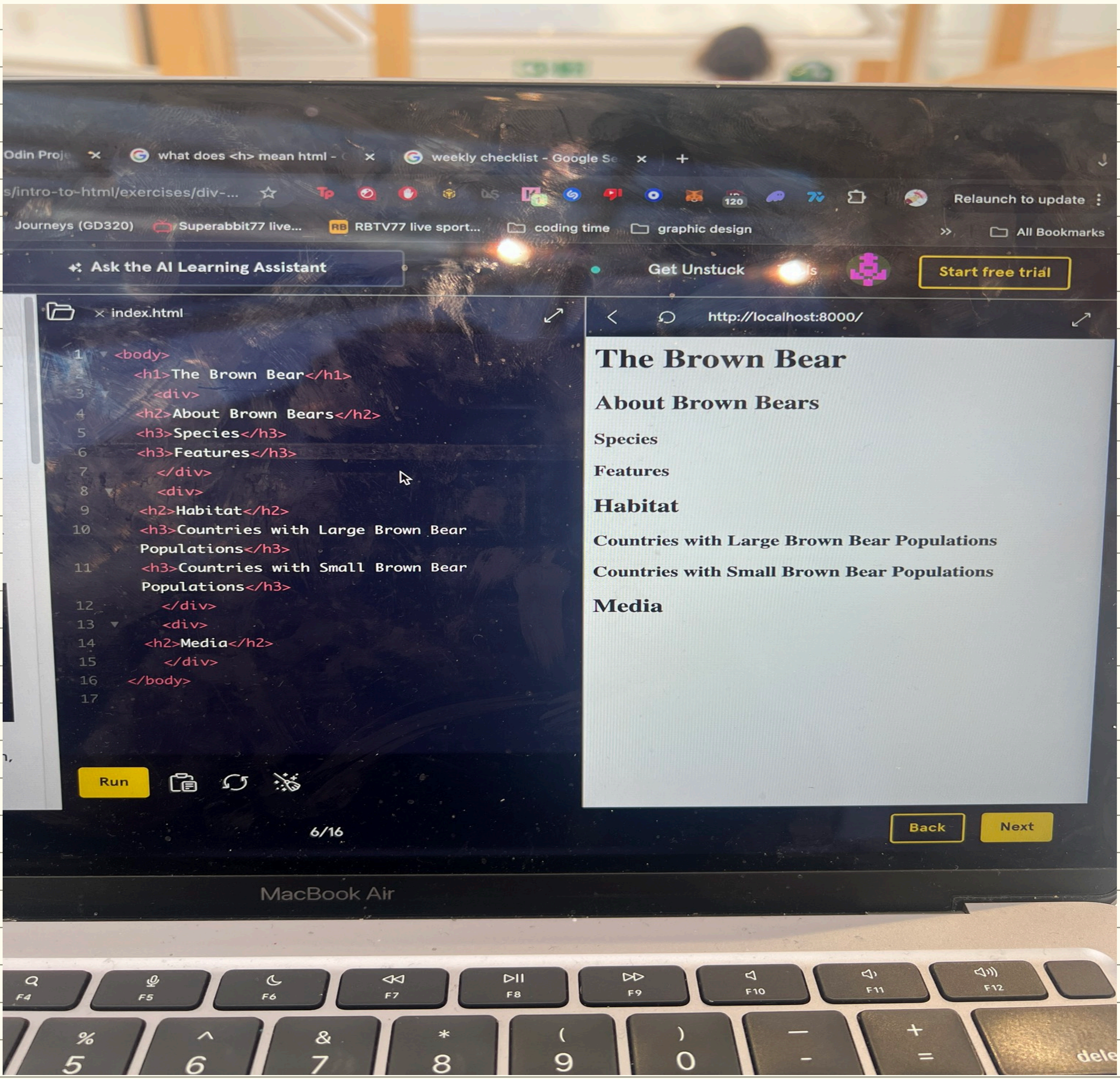
Countries with Large Brown Bear Populations

Countries with small Brown Bear Populations

Media

Divisions <div> </div>

- Useful for grouping elements in your HTML together.
- They don't inherently have visual representation, but they are very useful when we want to apply custom styles to our HTML elements.
- <div>s allow us to group HTML elements to apply the same styles for all HTML elements inside.
- <div>s can contain any text or other HTML elements, such as links, images or videos.
- Remember to always add 2 spaces of indentation when you nest elements inside of <div>s for better readability.



```
<body>
  <h1>The Brown Bear</h1>
    <div>
  <h2>About Brown Bears</h2>
  <h3>Species</h3>
  <h3>Features</h3>
    </div>
    <div>
  <h2>Habitat</h2>
  <h3>Countries with Large Brown Bear
  Populations</h3>
   <h3>Countries with Small Brown Bear
  Populations</h3>
    </div>
    <div>
  <h2>Media</h2>
    </div>
</body>
```

The Brown Bear

About Brown Bears

Species

Features

Habitat

Countries with Large Brown Bear Populations

Countries with Small Brown Bear Populations

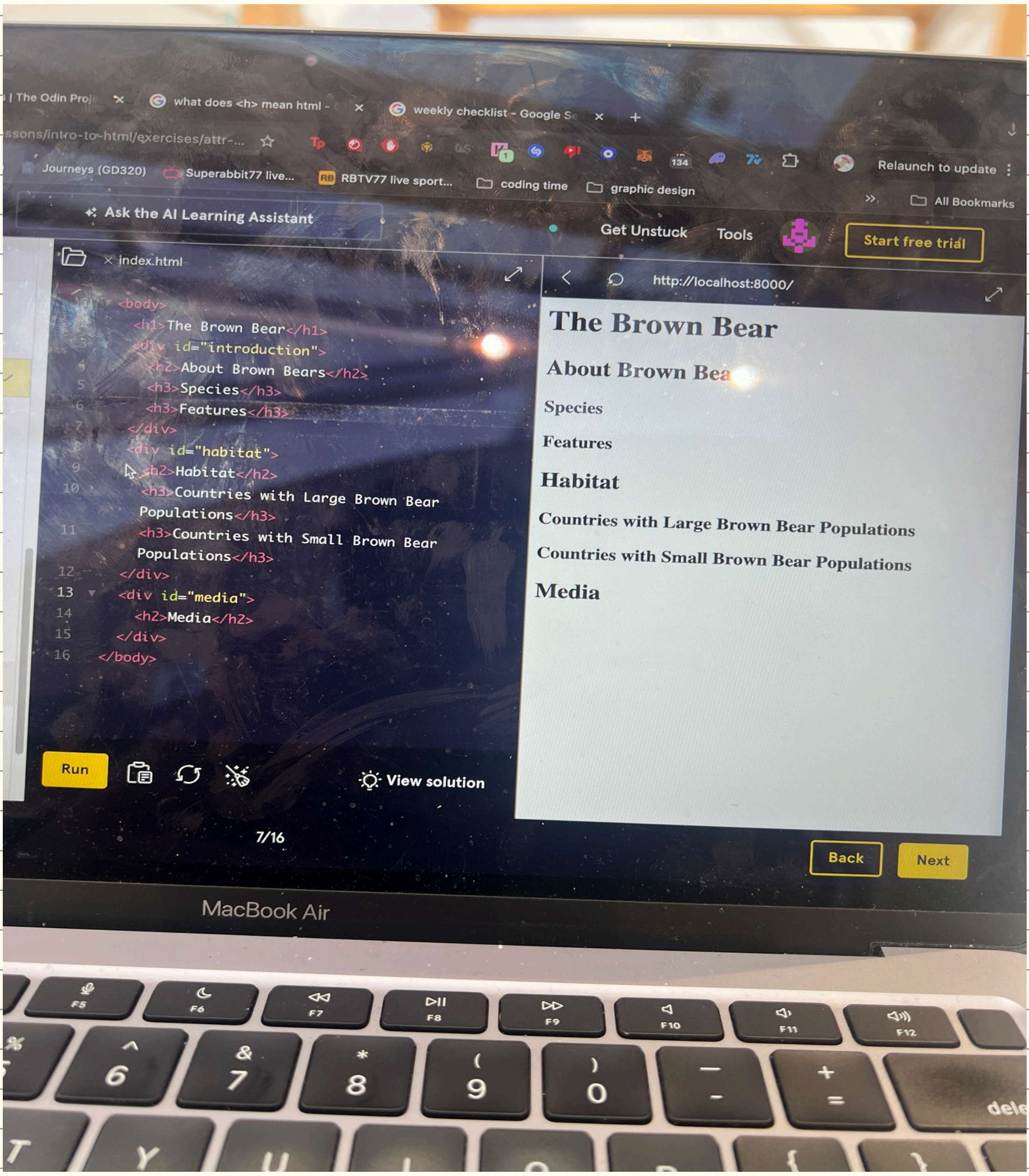
Media

what does <h> mean html -
weekly checklist - Google Se
s/intro-to-html/exercises/div-...
Relaunch to update
Journeys (GD320)
Superabbit77 live...
RBTV77 live sport...
coding time
graphic design
All Bookmarks
Ask the AI Learning Assistant
Get Unstuck
Start free trial
index.html
http://localhost:8000/
Run
6/16
Back
Next
MacBook Air

Attributes

- If you want to expand an elements tag, we can do so using an attribute.
- Attributes are content added to the opening tag of an element & can be used in several different ways, from providing information to changing styling.
- Attributes are made of the following 2 parts:
  - The name of the attribute
  - The value of the attribute.
- One commonly used attribute is the id. We can use the id attribute to specify different content (such as <div>s) and is really useful when you use an element more than once.
- Ids have several different purposes in HTML.

→
```
<div id="introduction">
    <h1> Introduction </h1>
</div>
```



index.html

```
<body>
  <h1>The Brown Bear</h1>
  <div id="introduction">
    <h2>About Brown Bears</h2>
    <h3>Species</h3>
    <h3>Features</h3>
  </div>
  <div id="habitat">
    <h2>Habitat</h2>
    <h3>Countries with Large Brown Bear Populations</h3>
    <h3>Countries with Small Brown Bear Populations</h3>
  </div>
  <div id="media">
    <h2>Media</h2>
  </div>
</body>
```

http://localhost:8000/

The Brown Bear

About Brown Bea

Species

Features

Habitat

Countries with Large Brown Bear Populations

Countries with Small Brown Bear Populations

Media

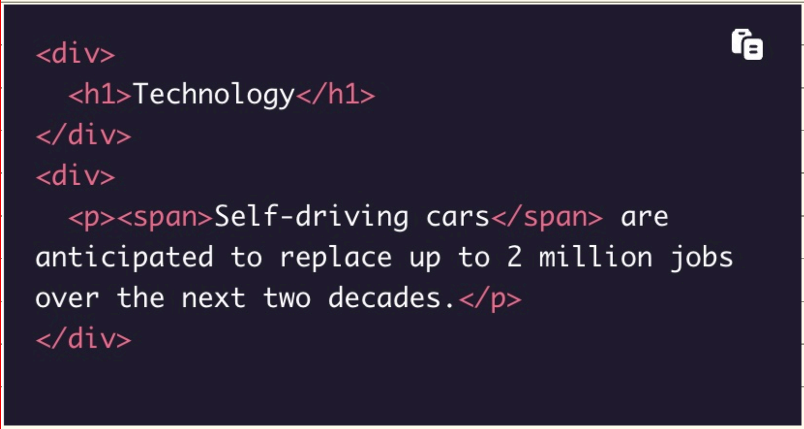
what does <h> mean html -
weekly checklist - Google S
Ask the AI Learning Assistant
Get Unstuck
Tools
Start free trial
Relaunch to update
Run
View solution
7/16
Back
Next
MacBook Air

Displaying text

- If you want to display text in HTML you can use a paragraph or a span:

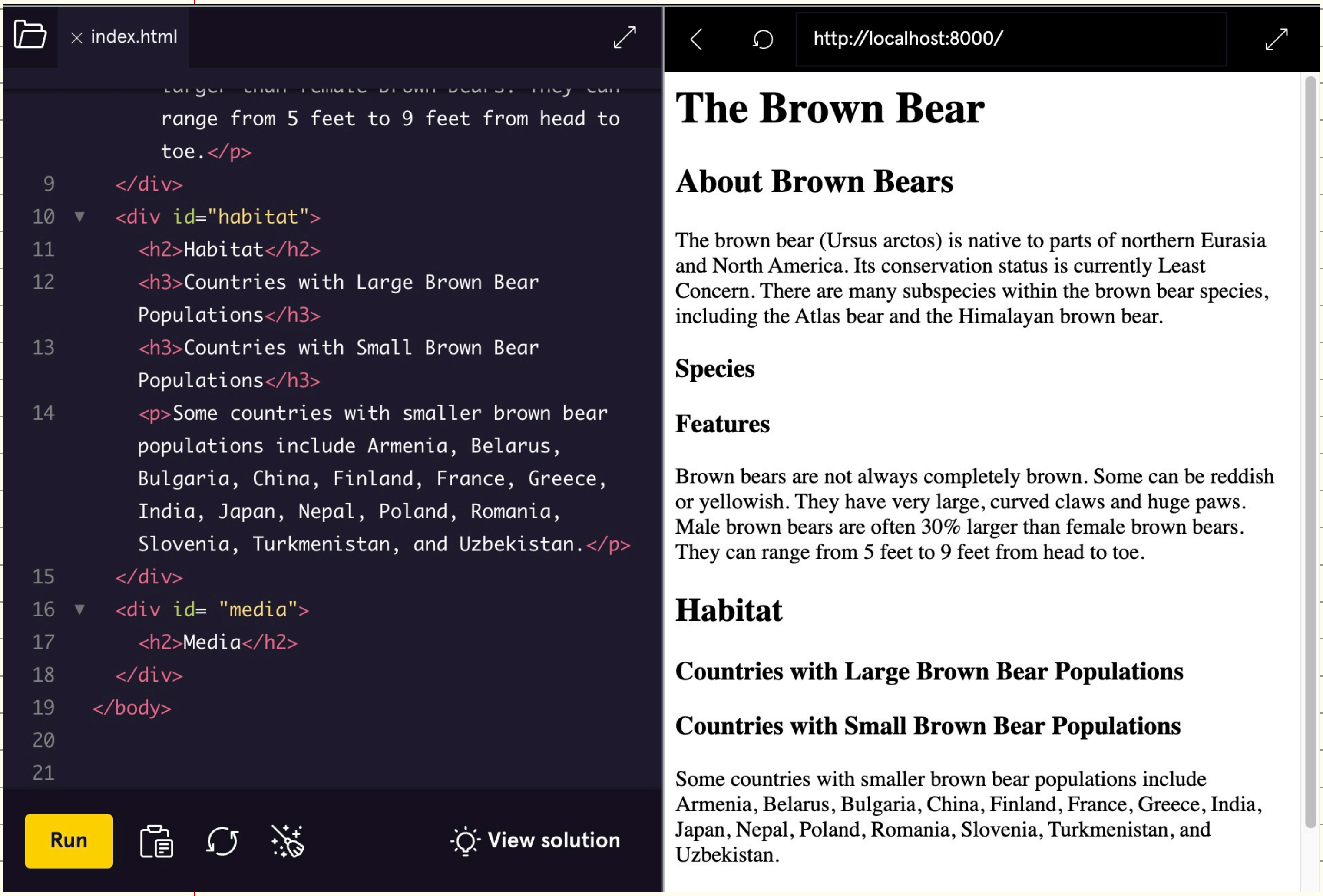
- Paragraphs (<p>) contain a plain block of text.

- <span> contains short pieces of text or other HTML. They are used to separate small pieces of content that are on the same line as other content.



```
<div>
  <h1>Technology</h1>
</div>
<div>
  <p><span>Self-driving cars</span> are
anticipated to replace up to 2 million jobs
over the next two decades.</p>
</div>
```

- The <span> element seperates "self driving cars" from the rest of the text in the paragraph.



```
      range from 5 feet to 9 feet from head to
      toe.</p>
    </div>
    <div id="habitat">
      <h2>Habitat</h2>
      <h3>Countries with Large Brown Bear
      Populations</h3>
      <h3>Countries with Small Brown Bear
      Populations</h3>
      <p>Some countries with smaller brown bear
      populations include Armenia, Belarus,
      Bulgaria, China, Finland, France, Greece,
      India, Japan, Nepal, Poland, Romania,
      Slovenia, Turkmenistan, and Uzbekistan.</p>
    </div>
    <div id= "media">
      <h2>Media</h2>
    </div>
  </body>
```

The Brown Bear

About Brown Bears

The brown bear (Ursus arctos) is native to parts of northern Eurasia and North America. Its conservation status is currently Least Concern. There are many subspecies within the brown bear species, including the Atlas bear and the Himalayan brown bear.

Species

Features

Brown bears are not always completely brown. Some can be reddish or yellowish. They have very large, curved claws and huge paws. Male brown bears are often 30% larger than female brown bears. They can range from 5 feet to 9 feet from head to toe.

Habitat

Countries with Large Brown Bear Populations

Countries with Small Brown Bear Populations

Some countries with smaller brown bear populations include Armenia, Belarus, Bulgaria, China, Finland, France, Greece, India, Japan, Nepal, Poland, Romania, Slovenia, Turkmenistan, and Uzbekistan.

Styling Text

25/1/25

- You can use the HTML tag <em> to emphasise text, while the <strong> highlights important text.
- Browsers have sheets that generally style these tags the following ways:
  - <em> = italics
  - <strong> = Bold

Line Breaks

- The line break element is unique because it's only composed of a starting tag. You can use it anywhere within your HTML code and a line break will be shown in the browser.

Unordered list <UL> </UL>

- In HTML you can use an unordered list (<ul>) to create a list of items in no particular order
- The <ul> element should not hold raw text and won't automatically format raw text into an unordered list of items. Individual list items must be added to the unordered list using the <li> tag. The <li> or list item tag is used to describe an item in a list.

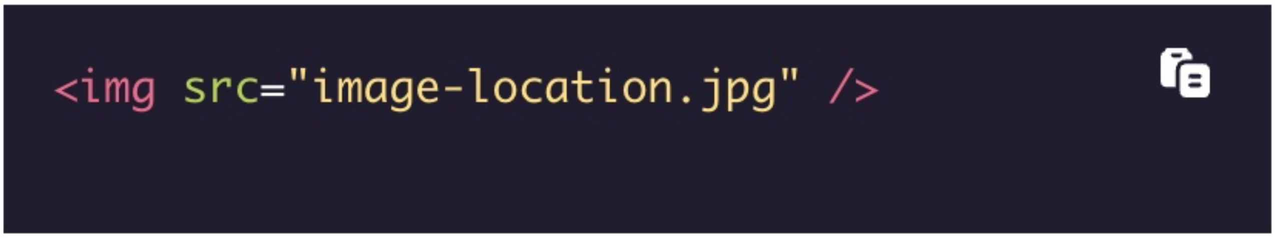
Ordered list <ol> </OL>

- Ordered lists are like unordered lists but are numbered instead.
- Created with <ol> tag and followed up with <li> tags.

Images

- The <img> tag allows you to add an image to a web page. Most elements require both opening / closing tags, but the <img> tag is a self-closing tag.
- The end of the <img> tag has a forward slash (/)

↓

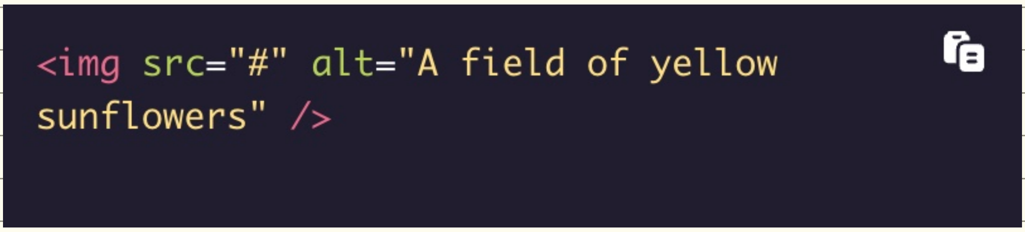


```
<img src="image-location.jpg" />
```

- The image tag has a required attribute called src.

Image Alts

- <alt> = alternative text
  → Brings meaning to the images on our sites.



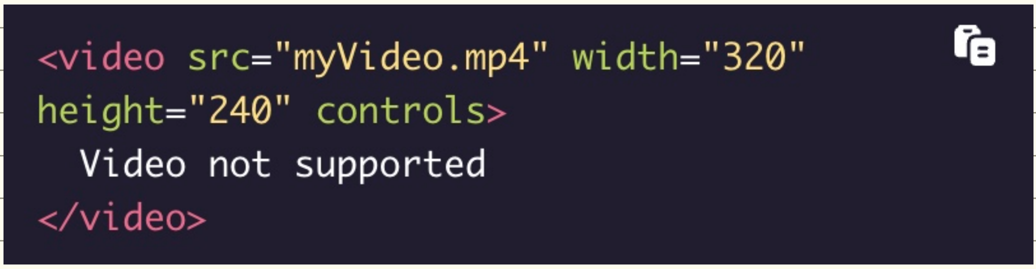
```
<img src="#" alt="A field of yellow
sunflowers" />
```

The alt attribute also serves the following purposes:

- If an image fails to load on a web page, a user can mouse over the area originally intended for the image & a brief description of the image
- Visually impaired users often browse the web with the aid of screen reading software. When you include the alt attribute, the screen reading software reads the alt attribute out loud.
- The alt attribute also plays a role in Search Engine Optimisation (SEO), because search engines cannot "see" the images on websites as they crawl the internet.

Videos <video>

- The <video> element also requires an src attribute with a link to the video source. Unlike the <img> element however, the <video> element requires a closing tag.
- After the src attribute, the width & height attributes are used to set the size of the video displayed in the browser.
- The controls attribute instructs the browser to include basic video controls such as pausing & playing.
- The text 'Video not supported' between opening & closing tags will only be displayed if the browser is unable to load the video



```
<video src="myVideo.mp4" width="320"
height="240" controls>
  Video not supported
</video>
```

Preparing for HTML

Learning how to set up a HTML file.

- HTML files require certain elements to set up the document correctly. We can let browsers know that we are using HTML by starting our document with a document type declaration.

The declaration looks like this:

`<!DOCTYPE html>`

- To add HTML structure and content, we must add opening & closing <html> tags after declaring <!DOCTYPE html>:

```
<!DOCTYPE html>
<html>


</html>
```

The Head

- Gives the browser information about the page itself.
- The <head> element contains the metadata for a web page. Metadata is information about the page that isn't displayed directly on the web page.

Page Titles

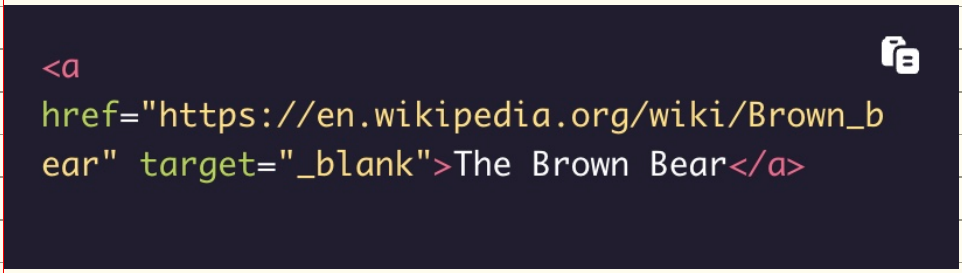
- The <title> tag is always inside the <head>.

Linking to other web pages <a>

- You can add links by adding an anchor element <a> and including the text of the link in between the opening & closing tags.
- It needs the [href] attribute. This attribute stands for hypertext reference & is used to link to a path.

Opening Links in a New window

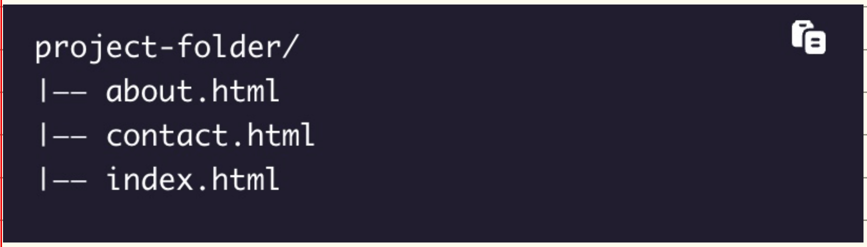
- target attribute specifies how a link should open.
- For a link to open in a new window, the target attribute requires a value of '_blank'
- The target attribute can be added directly to the opening tag of the anchor element, just like the href attribute.
- "_blank" attribute tells browser to open the relevant browser / tab.



```
<a
href="https://en.wikipedia.org/wiki/Brown_b
ear" target="_blank">The Brown Bear</a>
```

Linking to Relative Page

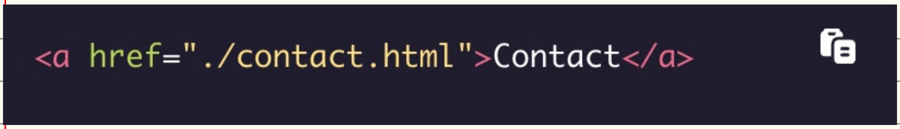
- When making multi-page static websites, web developers often store HTML files in the root directory, or a main folder where all the files for the project are stored.



```
project-folder/
|-- about.html
|-- contact.html
|-- index.html
```

3 Different files - about.html, contact.html & index.html.

- HTML files are often stored in the same folder. If the browser is currently displaying index.html it also knows about about.html & contact.html (as they're in the same folder).

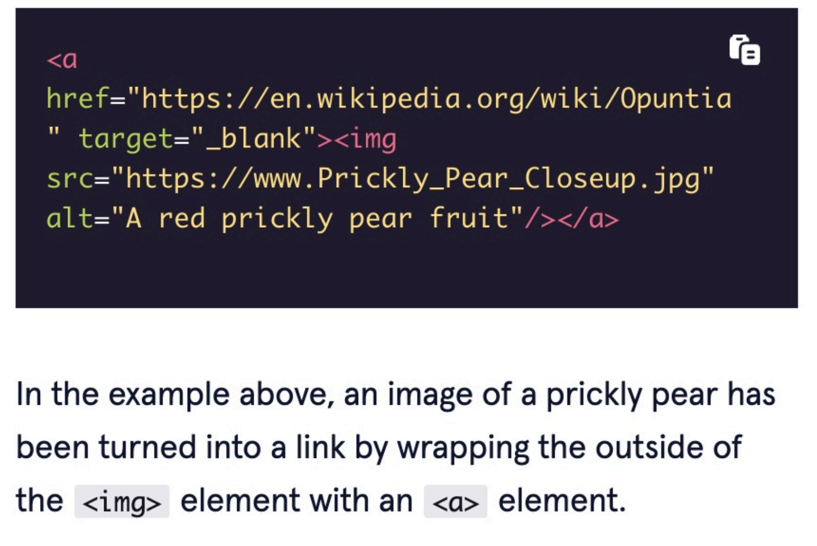


```
<a href="./contact.html">Contact</a>
```

<a> tag is used with a relative path to link from the current HTML file to the contact.html

- On the web page, Contact will appear as a link.
- the './' in "./index.html"

Linking at Will

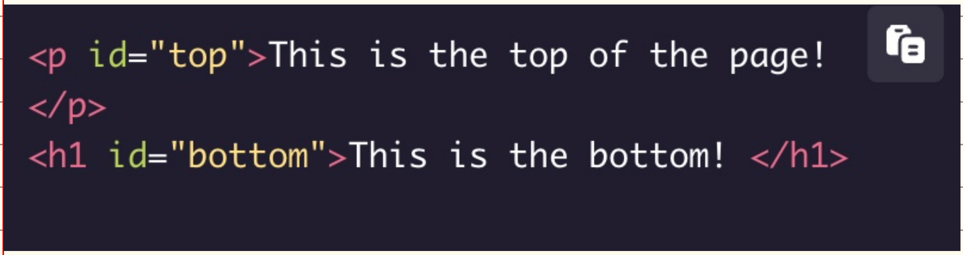


```
<a
href="https://en.wikipedia.org/wiki/Opuntia
" target="_blank"><img
src="https://www.Prickly_Pear_Closeup.jpg"
alt="A red prickly pear fruit"/></a>
```

In the example above, an image of a prickly pear has been turned into a link by wrapping the outside of the `<img>` element with an `<a>` element.

Linking to same page.

- In order to link to a target on the same page, we must give the target an id attribute & a unique value like this.



```
<p id="top">This is the top of the page!
</p>
<h1 id="bottom">This is the bottom! </h1>
```

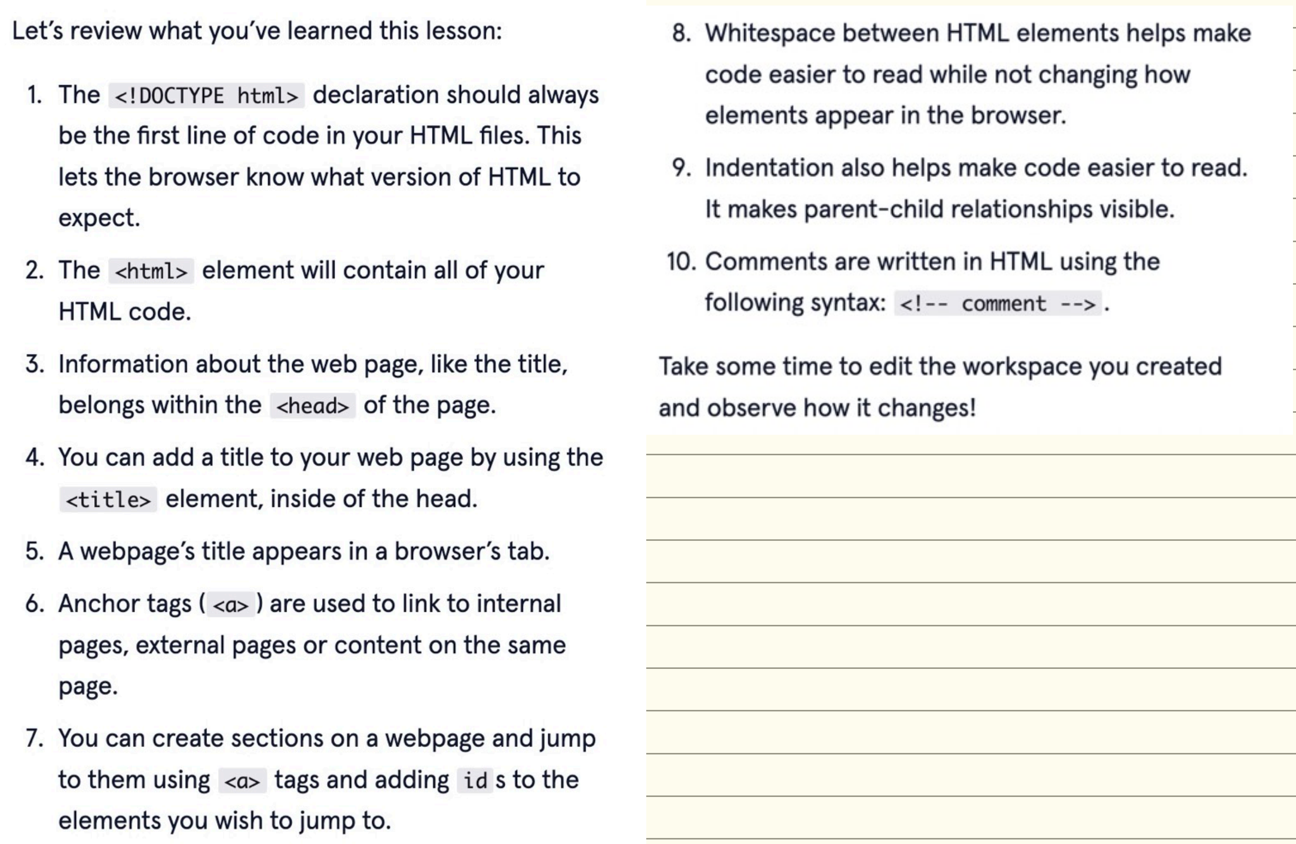
In this example the <p> is assigned an 'id' of top & the <h1> is assigned as "bottom"

Indentation

- If nested in <body>, indent 2 spaces.
- If nested in <div> indent 2spaces (additional if nested in body)

Comments

- Comments begin with "<!--" and end with "-->" any character between will be ignored by the browser.
- They allow you to experiment with code without needing to delete old code.



Let's review what you've learned this lesson:

1. The `<!DOCTYPE html>` declaration should always be the first line of code in your HTML files. This lets the browser know what version of HTML to expect.
2. The `<html>` element will contain all of your HTML code.
3. Information about the web page, like the title, belongs within the `<head>` of the page.
4. You can add a title to your web page by using the `<title>` element, inside of the head.
5. A webpage's title appears in a browser's tab.
6. Anchor tags (`<a>`) are used to link to internal pages, external pages or content on the same page.
7. You can create sections on a webpage and jump to them using `<a>` tags and adding `id`s to the elements you wish to jump to.
8. Whitespace between HTML elements helps make code easier to read while not changing how elements appear in the browser.
9. Indentation also helps make code easier to read. It makes parent-child relationships visible.
10. Comments are written in HTML using the following syntax: `<!-- comment -->`.

Take some time to edit the workspace you created and observe how it changes!

Fashion Blog Brief (notes)

- & = isn't allowed in html so use "&amp;" to get passed the error message.

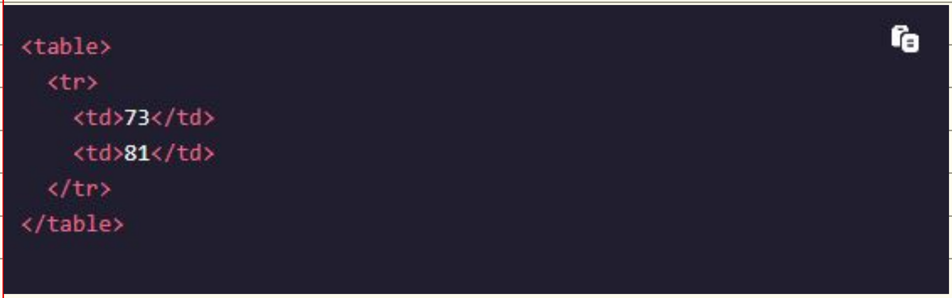
Tables <table></table>

- In many programs the table is predefined meaning it contains the rows, columns and cells that will hold data.

Table Rows <tr> </tr>

Table Data <td> </td>

- Rows aren't enough to add data to a table. Each cell element must also be defined.
- In html you can add data using the table data element: <td>

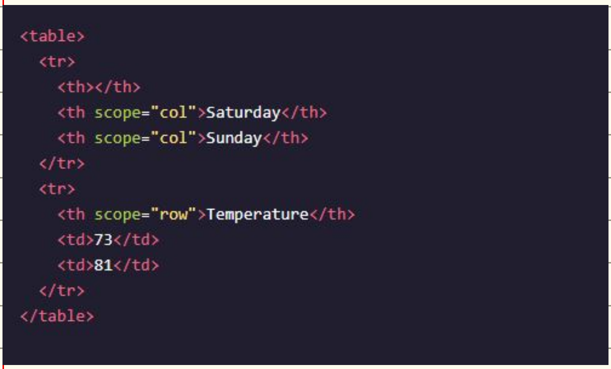


```
<table>
  <tr>
    <td>73</td>
    <td>81</td>
  </tr>
</table>
```

In this example, 2 data points (73 & 81) were entered into one row. By adding 2 data points, you create 2 cells of data.

Table Headings <th> </th>

- Must be placed within a table row



```
<table>
  <tr>
    <th></th>
    <th scope="col">Saturday</th>
    <th scope="col">Sunday</th>
  </tr>
  <tr>
    <th scope="row">Temperature</th>
    <td>73</td>
    <td>81</td>
  </tr>
</table>
```

- The blank heading allows the extra table cell necessary to allign the table headings correctly over the data they correspond to.
- row : makes it clear the headings for the row
- col : makes it clear heading is for column

Table Borders

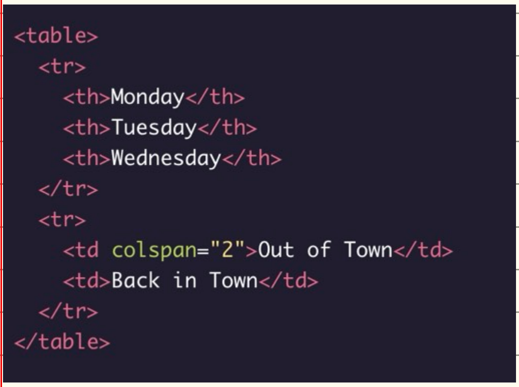
- We use CSS to add style to HTML documents, because it helps to seperate the structure of the page from how it looks.

Spanning Columns colspan

For when the table contains data that spans multiple columns.

- ↳ E.g, a colender, that spans across multiple hours/days.

- Data can span columns using the colspan attribute. The attribute accepts an integer (greater than or equal to 1)



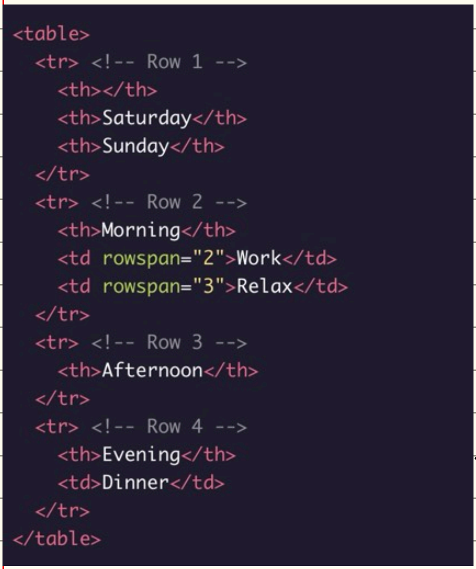
```
<table>
  <tr>
    <th>Monday</th>
    <th>Tuesday</th>
    <th>Wednesday</th>
  </tr>
  <tr>
    <td colspan="2">Out of Town</td>
    <td>Back in Town</td>
  </tr>
</table>
```

In this example above, the data 'Out of town' spans the Monday & Tuesday table headings using the value 2 (2 columns).

The data "Back in town" appears only under the Wednesday heading

Spanning Rows rowspan

- Data can also span multiple rows using the rowspan attribute.



```
<table>
  <tr> <!-- Row 1 -->
    <th></th>
    <th>Saturday</th>
    <th>Sunday</th>
  </tr>
  <tr> <!-- Row 2 -->
    <th>Morning</th>
    <td rowspan="2">Work</td>
    <td rowspan="3">Relax</td>
  </tr>
  <tr> <!-- Row 3 -->
    <th>Afternoon</th>
  </tr>
  <tr> <!-- Row 4 -->
    <th>Evening</th>
    <td>Dinner</td>
  </tr>
</table>
```

In this example, there are 4 rows:

* The first row contains an empty cell & the 2 column headings.

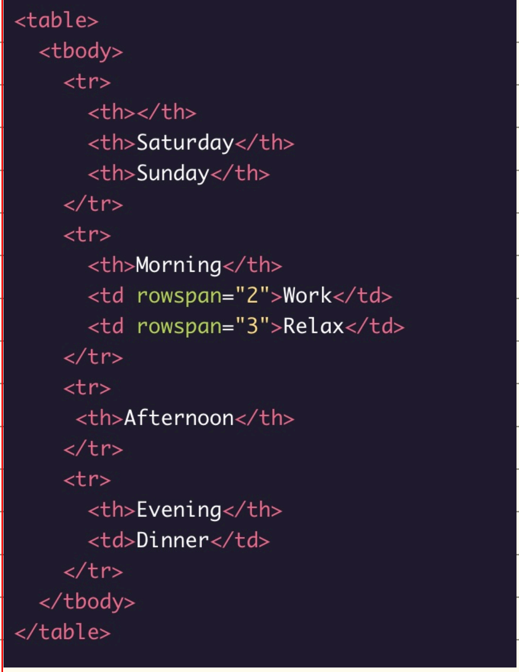
* The second row contains the morning row heading, along with work, which spans 2 rows under the Saturday column. The "Relax" entry spans 3 rows under the Sunday column.

* The 3rd row only contains the afternoon row heading.

* The 4th row only contains the Dinner entry, since "Relax" spans into the cell next to it.

Table Body <tbody>

- Over time, a table can grow to contain a lot of data & become very long. When this happens, the table can be sectioned off with the <tbody> element to make it easier to manage.

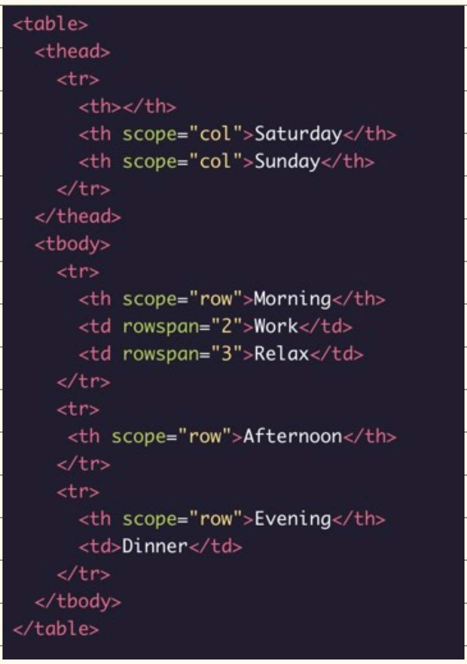


```
<table>
  <tbody>
    <tr>
      <th></th>
      <th>Saturday</th>
      <th>Sunday</th>
    </tr>
    <tr>
      <th>Morning</th>
      <td rowspan="2">Work</td>
      <td rowspan="3">Relax</td>
    </tr>
    <tr>
     <th>Afternoon</th>
    </tr>
    <tr>
      <th>Evening</th>
      <td>Dinner</td>
    </tr>
  </tbody>
</table>
```

- In this example all of the table data is contained within a table body element.

Table Head <thead>

- When a table's body is sectioned off, however, it also makes sense to section off the tables column headings using the <thead> element.



```
<table>
  <thead>
    <tr>
      <th></th>
      <th scope="col">Saturday</th>
      <th scope="col">Sunday</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th scope="row">Morning</th>
      <td rowspan="2">Work</td>
      <td rowspan="3">Relax</td>
    </tr>
    <tr>
     <th scope="row">Afternoon</th>
    </tr>
    <tr>
      <th scope="row">Evening</th>
      <td>Dinner</td>
    </tr>
  </tbody>
</table>
```

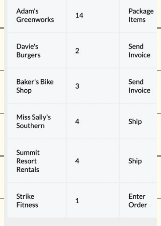
In this example above, the only new element is the <thead>

Additionally, only the column headings go under the <thead> element. We can use the scope attribute on <th> elements to indicate whether a <th> element is being used as a "row" heading or a "col" heading.

Table Footer <tfoot>

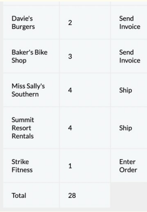
- The bottom part of a long table can also be sectioned off using the <tfoot> element.

- In the example below, the footer contains the totals of the data in the table. Footers are often used to contain sums, differences & other data results.



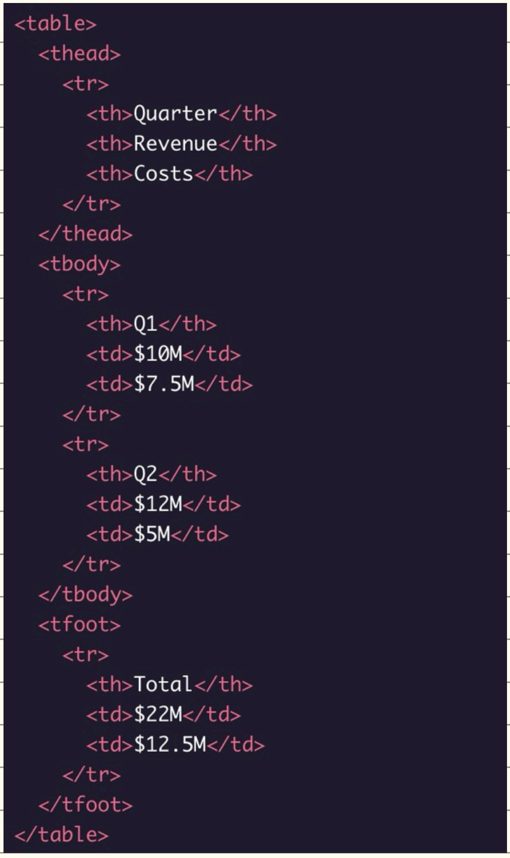
Adam's Greenworks	14	Package Items
Davie's Burgers	2	Send Invoice
Baker's Bike Shop	3	Send Invoice
Miss Sally's Southern	4	Ship
Summit Resort Rentals	4	Ship
Strike Fitness	1	Enter Order

Without Footer (Before)



Davie's Burgers	2	Send Invoice
Baker's Bike Shop	3	Send Invoice
Miss Sally's Southern	4	Ship
Summit Resort Rentals	4	Ship
Strike Fitness	1	Enter Order
Total	28	

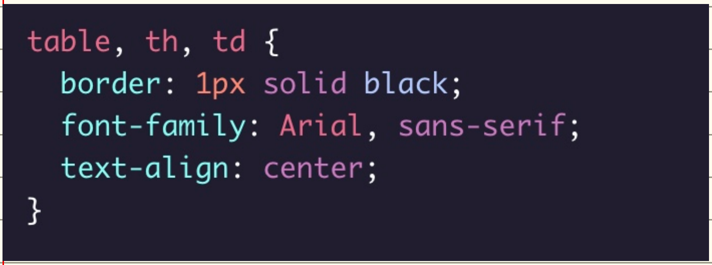
With Footer (After)



```html
<table>
  <thead>
    <tr>
      <th>Quarter</th>
      <th>Revenue</th>
      <th>Costs</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>Q1</th>
      <td>$10M</td>
      <td>$7.5M</td>
    </tr>
    <tr>
      <th>Q2</th>
      <td>$12M</td>
      <td>$5M</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <th>Total</th>
      <td>$22M</td>
      <td>$12.5M</td>
    </tr>
  </tfoot>
</table>
```

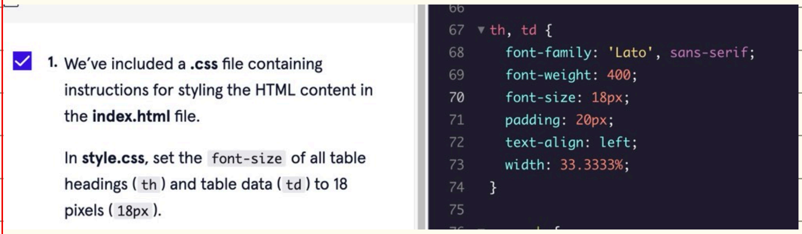
Styling with CSS

- Tables by default are very bland. They have no borders, the font colour is black, & the typeface is the same type used for other HTML elements.
- You can use CSS (cascading style sheets) to style tables.



```css
table, th, td {
  border: 1px solid black;
  font-family: Arial, sans-serif;
  text-align: center;
}
```

- The code in the example above demonstrates just some of the various table aspects you can style using CSS properties.



☑ 1. We've included a **.css** file containing instructions for styling the HTML content in the **index.html** file.

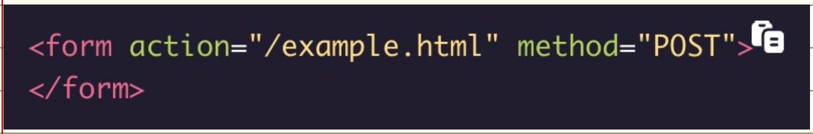
In **style.css**, set the `font-size` of all table headings (`th`) and table data (`td`) to 18 pixels (`18px`).

```css
th, td {
  font-family: 'Lato', sans-serif;
  font-weight: 400;
  font-size: 18px;
  padding: 20px;
  text-align: left;
  width: 33.3333%;
}
```

How a form works

<form> action & method

- The <form> element is a great tool for collecting information

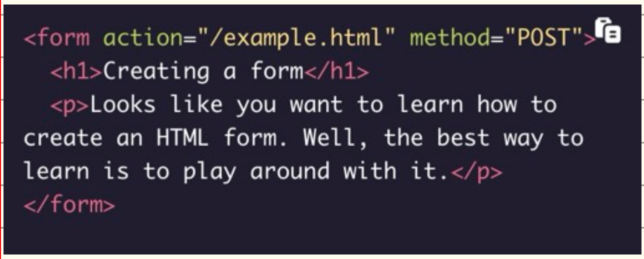


```
<form action="/example.html" method="POST">
</form>
```

- In this example, it's created a skeleton for a <form> that'll send information to example.html as a post request.
  - ↳ The action attribute determines where the information is sent.
  - ↳ The method attribute is assigned a HTTP verb that is included in the HTTP Request.

- (NOTE: HTTP verbs like POST do not need to be capatalised for the request to work, but is done out of convention.)

- The form element can also contain child elements. For instance, it would be helpful to provide a header so that users know what the <form> is about. You could also add a paragraph to provide even more detail, like this example:



```
<form action="/example.html" method="POST">
  <h1>Creating a form</h1>
  <p>Looks like you want to learn how to
create an HTML form. Well, the best way to
learn is to play around with it.</p>
</form>
```

- This example doesn't collect any user input.

Text Input

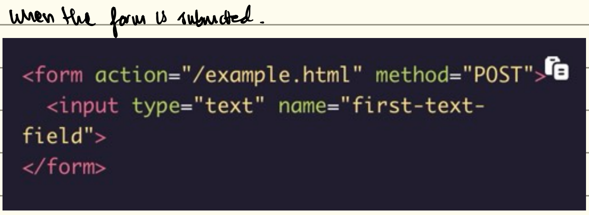
<input> type & name & value

- If we want to allow people to input in our <form>, we'll need the help of the <input> element.

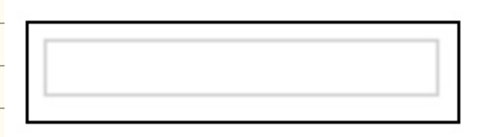
- The input element has a type of attribute which determines how it renders on the web page & what kind of data it can accept.

- "text". – When we create an <input> element with type ="text", it renders a field for users to type onto. (NOTE: the default value of type is "text".

- Its also important that you include a name attribute for the <input>. Without it information in the input won't be sent when the form is submitted.

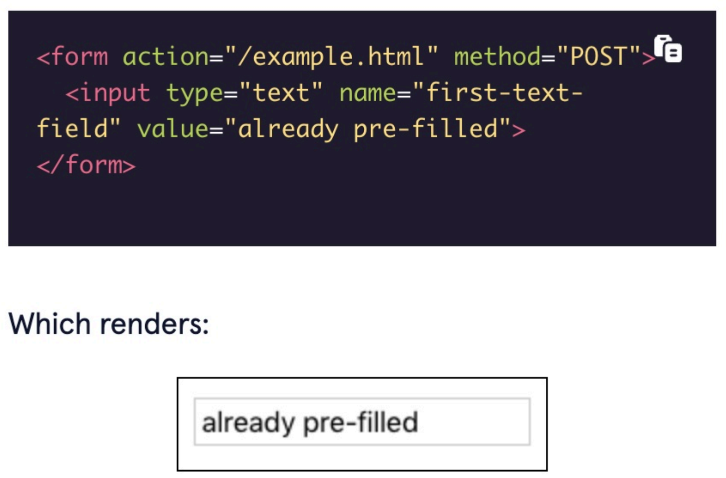


```
<form action="/example.html" method="POST">
  <input type="text" name="first-text-
field">
</form>
```



an empty box. (for chrome browser)

- After users type into the <input> element the value of the value attribute becomes what is typed into the text field. The value of the value attribute is paired with the value of the name attribute & send as ~~text~~ when the form is submitted.
  - ↳ Value = Data put into the text.



```
<form action="/example.html" method="POST">
  <input type="text" name="first-text-
field" value="already pre-filled">
</form>
```

Which renders:

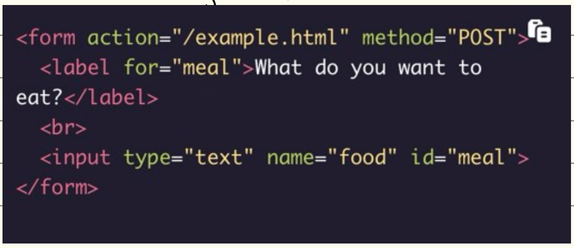
already pre-filled

- Example

Adding a label <label>

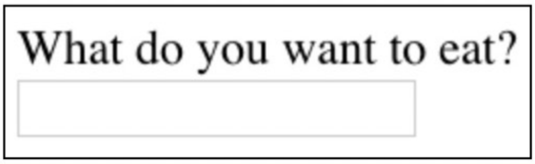
- For users to correctly identify an <input> we use the appropriately named <label> element.

- To associate a <label> and an input, the input needs an id attribute.
  - Then assign the for attribute of the label element with the value of the id attribute of <input> like this:



```
<form action="/example.html" method="POST">
  <label for="meal">What do you want to
eat?</label>
  <br>
  <input type="text" name="food" id="meal">
</form>
```

which = renders

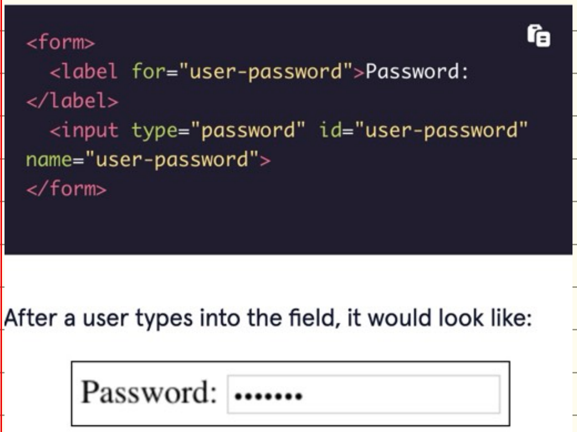


What do you want to eat?

Password Input

- There is a type = "password" attribute for <input>.

- An <input type ="password"> element will replace input text with another character like an asterisk (*) or a dot (·). Example of how to create a password field:



```
<form>
  <label for="user-password">Password:
</label>
  <input type="password" id="user-password"
name="user-password">
</form>
```

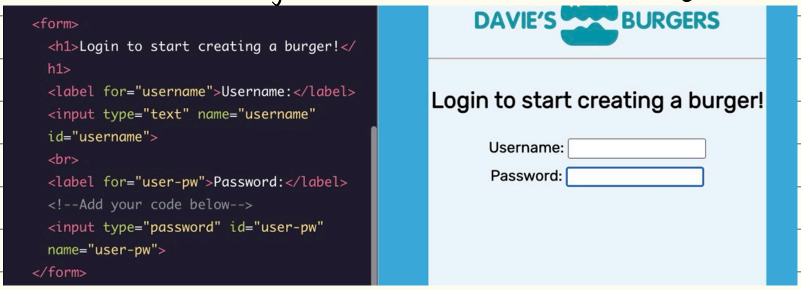
After a user types into the field, it would look like:

Password: •••••••

· Even though the password obscures the text, when the form is sent, so is the value of the text.

· In other words, if "CJ-123" is typed into the password field, "user-password=CJ-123" is sent along with the other information on the form.

(my example from the class I'm doing)



```
<form>
  <h1>Login to start creating a burger!</
h1>
  <label for="username">Username:</label>
  <input type="text" name="username"
id="username">
  <br>
  <label for="user-pw">Password:</label>
  <!--Add your code below-->
  <input type="password" id="user-pw"
name="user-pw">
</form>
```

Login to start creating a burger!

Username:

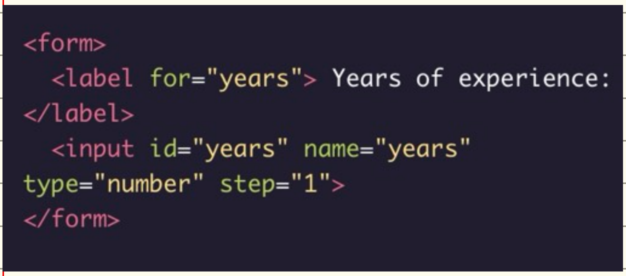
Password:

Number Input. type = "number" step

- By setting type="number" for an <input> we can restrict what users type into the input field to just numbers (and special characters '-', '+' & '.').

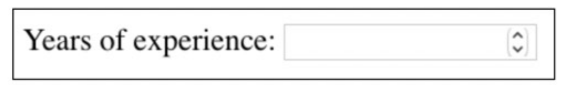
- We can also use the step attribute which creates arrows inside the input field to increase or decrease by the value of the step attribute

- Example code needed to render an input field for numbers:



```
<form>
  <label for="years"> Years of experience:
</label>
  <input id="years" name="years"
type="number" step="1">
</form>
```

=



Years of experience: